

LA 99ers

# TOPICS

Vol 11 No 12

Dec 1992

Season's Greetings  
 To The Entire  
 TI Community  
 From The LA99ers

\*\*\*\*\*  
 \*\*\*\*\*

In This Issue

Convert-The Program.....	By Earl Raguse...	Pg 1
TI Writer Tip .....	From Nutmeg News.	Pg 6
TI World News .....	By J. Peterson	Pg 7
3.5in Disk Drives .....	By F. RyIstock	Pg 8
LA 99er Library .....	By Sy Silver	Pg 9
Officers .....		Pg 10

\*\*\*\*\*  
 \*\*\*\*\*

Thanks to all those who participated  
 in the meetings and contributed to  
 the newsletter. Next Meeting Jan 13, 1993

HAPPY NEWYEAR ALL

\*\*\*\*\*



CONVERT -- The Program

Last month I talked about the program called CONVERT, and I promised to print it so you could type it in for yourself. But if you are just plain incapable of typing in a program, you may obtain it directly from me, or the group library. You will learn a lot more however if you type it in and try to understand each step yourself. I will explain what goes on, not completely step by step, that would be too long, but I will detail what each group of steps is doing. If you have trouble, call me. You will recall, unless this is your first exposure, that CONVERT is a menu driven program that can read any type of data file, not program files, but it does not guarantee that you can make sense of it, but you CAN read it, OR you may rewrite the file to another format. It is frequently useful to rewrite a file to DV80 format, (Display Variable, 80 max record length) so you may read and edit it with TIW/FW. I have not seen a general purpose program for doing this. For any individual file, it is very easy to write a program to do this. That is, if you know how, of course. A general menu driven program is less easy. CONVERT was started a some time ago in the 90's, and had problems, that I initially wrote about in the Feb 92 ROM. I will not repeat any of that. I had not yet at that time made an extensive use of subprograms, so it is now a mixture of brute force coding and subprograms. The subprograms used herein begin at line 1000. This program also makes use of a lot of subroutines. They start at line 540, and end at line 860. Lines 870-950 are the in-program docs. The program actually ends at line 530, where all files are closed, and the program returns to the start up menu, line 120, where you may Quit, read the Docs, Proceed to do another file or whatever. That is a good place to start. When you insert my XB#8 disk, and select Extended Basic, a DIRectory (menu) loads, where you may press A for A Converter. CONVERT then loads, and line 120 displays the name and 130 shows a menu to select from. I advise you to select

```

100 ! SAVE DSK1.CONVERT
110 ! By Earl Raguse 9/90
      Revised 5/92
120 DISPLAY AT(02,1)ERASE ALL:" FILE CONVERSION PROGRAM
" :: IRL=80 :: ORL=80
130 DISPLAY AT(5,1):"      Se
lect First Letter": "
      Instruction
      P Proceed
      Q Quit"
140 CALL GKEY(K,24):: IF K=A
SC("I")THEN B70 ELSE IF K=AS
C("Q")THEN RUN "DSK1.DIR
"
150 DISPLAY AT(10,1)ERASE AL
L:"      Select First Letter
": "      Read a Fi
le      Convert F
ile" :: RF=0
160 CALL GO("CR",K):: ON K G
OTO 180,170
170 RF=1
180 DISPLAY AT(12,1)ERASE AL
L:"Input File Name": "
DSK1."
190 ACCEPT AT(14,9)SIZE(-12)
:IFS :: IDNS="DSK"ASEGS(IFS,
1,2):: IFS=SEGS(IFS,3,LE
N(IFS)-2)
200 GOSUB 550 :: GOSUB 580 :
: CALL GKEY(K,24):: IF K=ASC
("1")THEN ID=4 ELSE IF K
=ASC("2")THEN ID=0
210 IF ID=4 THEN IORGNS="REL
," ELSE IORGNS="SEQ,"
220 GOSUB 550 :: GOSUB 590 :
: CALL GKEY(K,24):: K=K-48 :
: IF K=1 THEN IT=2 ELSE
IF K=2 THEN IT=0
230 IF IT=2 THEN ITPES="DIS
," ELSE ITPES="INT,"
240 GOSUB 550 :: GOSUB 600 :
: CALL GKEY(K,24):: K=K-48 :
: IF K=1 THEN IK=1 ELSE
IF K=2 THEN IK=0
250 IF IK=1 THEN IKINDS="VAR
" ELSE IKINDS="FIX"
260 GOSUB 950 :: IRL=RL
270 IF ID=4 THEN IKINDS="FIX
"
280 DISPLAY AT(12,1)ERASE AL
L:IDNS;IFS;","INPUT,","IORGNS;
ITPES;IKINDS;IRL
290 DISPLAY AT(20,1):"IS THI
S CORRECT Y/N "
300 CALL GO("NY",K):: ON K G
OTO 180,310

```

Instruction the first time you run it. That will take you to line 870 where lines 870-940 print a screen telling you a little about CONVERT. Line 950 invokes the subprogram PAK. PAK is nothing more than a stopping off place in a program, that waits until you press a key, as instructed by the message at the bottom of the screen. PAK is programmed on lines 1120-1150. When it returns to line 950, GOTO 120 is executed, and you are back where you started from.

Line 140 will invoke subprogram GKEY (see line 1010-30) it examines your key stroke, and sends the program to the appropriate place. Assuming that you select P for Proceed, the program will go to line 150, where you will get a new set of choices, (this is the latest revision) either Read a File or Convert a File. It also sets RF=0. Line 160 invokes another subprogram, GO, a very efficient way of handling a menu choice, I should have used GO for line 140, but I hadn't written it yet at that time.

If you select Read, the program goes to line 170 where RF is set to 1 (RF=Read File), and then on to 180. If you select Convert, the program goes directly to line 180, and RF is 0 as set in line 150. RF is used as a decision element later in line 310 to CALL READIT, a subroutine, see lines 1160-1210, that reads and displays the entire file.

Lines 180-280 find out the name and the input file parameters, which are stored in variables IO, IT, and IK, meaning, Input Organization, Input Type, and Input Kind designations for the file, then displays it all on the screen for your checking. I will explain about that later. Line 290 asks you to verify it.

On line 300, GO is invoked and if you select N for NO you will be returned to line 180 and allowed to correct any error you made, without having to retype it all. If you verify by typing Y, the program will go to 310 where if you had selected Read, READIT is invoked as explained above, after line 310 does a GOSUB 810 to open the file. Lines 810-83 is the place where IO, IT, and IK are summed up, and the sum plus 1 is used in and ON GOSUB statement to go to

```

310 GOSUB 810 :: IF RF=1 THE
N CALL READIT(1):: GOTO 150
320 DISPLAY AT(12,1)ERASE AL
L:"Output File Name": "
DSK1."
330 ACCEPT AT(14,9)SIZE(-15)
:DFS :: ODN$="DSK"&SEGS(DFS,
1,2):: DFS=SEGS(DFS,3,LE
N(DFS)-2)
340 GOSUB 560 :: GOSUB 580 :
: CALL GKEY(K,24):: IF K=ASC
("1")THEN OO=4 ELSE IF K
=ASC("2")THEN OO=0
350 IF OO=4 THEN ODRGNS="REL
," ELSE ODRGNS="SEQ,"
360 GOSUB 560 :: GOSUB 590 :
: CALL GKEY(K,24):: IF K=ASC
("1")THEN OT=2 ELSE IF K
=ASC("2")THEN OT=0
370 IF OT=2 THEN OTYPES="DIS
," ELSE OTYPES="INT,"
380 GOSUB 560 :: GOSUB 600 :
: CALL GKEY(K,24):: IF K=ASC
("1")THEN OK=1 ELSE IF K
=ASC("2")THEN OK=0
390 IF OK=1 THEN OKINDS="UAR
" ELSE OKINDS="FIX"
400 GOSUB 960 :: ORL=RL
410 IF OO=4 THEN OKINDS="FIX
"
420 DISPLAY AT(12,1)ERASE AL
L:ODN$;DFS;"",OUTPUT,"";ODRGNS
;;OTYPES;OKINDS;ORL
430 DISPLAY AT(20,1):"IS THI
S CORRECT Y/N "
440 CALL GO("NY",K):: ON K G
OTO 320,450
450 GOSUB 840
460 DISPLAY AT(20,1):"Readin
g & Writing Record "
470 N=0
480 IF EOF(1)THEN 530
490 INPUT #1:AS :: CALL CLSC
(22,24):: DISPLAY AT(22,1):AS
500 IF ASC(AS)>127 THEN 530
510 PRINT #2:AS
520 N=N+1 :: DISPLAY AT(20,2
6):N :: GOTO 480
530 CLOSE #1 :: CLOSE #2 ::
GOTO 120 :: END
540 ! **file designation**
550 DISPLAY AT(09,6)ERASE AL
L:"For the Input File" :: RE
TURN
560 DISPLAY AT(09,6)ERASE AL
L:"For the Output File" :: R
ETURN

```

the proper line of open statements, 620-700 to open the input file. The same sort of logic happens with the output file, on line 840-860 where variables OD, OT, and OK are summed to direct the ON GOSUB command to the appropriate line 730-800 to open the output file. This is the scheme that I mentioned in the Feb 92 ROM, when I found I could not use strings to enter the file parameters. Once the input file is open and you have selected to Read it, in which case we do not need an output file. After the file is read, we return to the menu of 150 again. Else, we must now open an output file. The output file parameters are requested of and input by you on lines 320-440 and the file opened on line 450 with GOSUB 840 as explained above. The input file is handled in the same way so I will not explain that. Lines 320-330 get the output file drive number ODN\$, and the output file name OF\$. Line 340 does a GOSUB 560 and 580. These two subroutines place a menu on the screen for Output file Organization, after your selections are recorded and displayed by GKEY (see lines 1010), the proper value is inserted in variable OD, and the correct text is put into OORGN\$ for later display. Lines 360-370 does the same sort of thing with subroutines 560 and 590 for Output file Type. The appropriate values are put into OT and OTYPE\$. Lines 380-390 do it again with subroutines 560 and 600, for Output file Kind, and the variables OK and OKIND\$. Lines 400 and 410 use subroutine 960 to get the Record length, RL, for the Output file, which is stored in ORL. In addition if OD=4, meaning that this is a RELATIVE file, then OKIND\$ is changed to "FIX". Line 420 displays the file characteristics you entered and lines 430 and 440 ask you verify it. If you enter N, you are returned to 320 to fix it. If you do verify it by entering Y, line 450 does a GOSUB 840, to open the file as described above. Lines 460-530 actually read a record from the input file, and write a record to the output file. Lines 460-480 place a message on the screen, set N=0 and

```

570 !**display menu**
580 DISPLAY AT(12,1):"Choose
:                               File
   1. Relative                   Or
gani 2. Sequential              za
tion" :: RETURN
590 DISPLAY AT(12,1):"Choose
:                               File
   1. Display                    Ty
pe 2. Internal" :: RETURN
600 DISPLAY AT(12,1):"Choose
:                               Record
   1. Variable                   Ki
nd 2. Fixed" :: RETURN
610 ! ***Open Input File***
620 OPEN #1:IDNS&IFS,SEQUENT
IAL,INTERNAL,FIXED IRL,INPUT
:: RETURN
630 IRL=100
640 OPEN #1:IDNS&IFS,SEQUENT
IAL,INTERNAL,VARIABLE IRL,IN
PUT :: RETURN
650 OPEN #1:IDNS&IFS,SEQUENT
IAL,DISPLAY ,FIXED IRL,INPUT
:: RETURN
660 OPEN #1:IDNS&IFS,SEQUENT
IAL,DISPLAY ,VARIABLE IRL,IN
PUT :: RETURN
670 OPEN #1:IDNS&IFS,RELATIV
E,INTERNAL,FIXED IRL,INPUT :
: RETURN
680 OPEN #1:IDNS&IFS,RELATIV
E,INTERNAL,FIXED IRL,INPUT :
: RETURN
690 OPEN #1:IDNS&IFS,RELATIV
E,DISPLAY ,FIXED IRL,INPUT :
: RETURN
700 OPEN #1:IDNS&IFS,RELATIV
E,DISPLAY ,FIXED IRL,INPUT :
: RETURN
710 !
720 ! ***Open Output File***
730 OPEN #2:ODNS&OFS,SEQUENT
IAL,INTERNAL,FIXED ORL,OUTPU
T :: RETURN
740 OPEN #2:ODNS&OFS,SEQUENT
IAL,INTERNAL,VARIABLE ORL,OU
TPUT :: RETURN
750 OPEN #2:ODNS&OFS,SEQUENT
IAL,DISPLAY ,FIXED ORL,OUTPU
T :: RETURN
760 OPEN #2:ODNS&OFS,SEQUENT
IAL,DISPLAY ,VARIABLE ORL,OU
TPUT :: RETURN
770 OPEN #2:ODNS&OFS,RELATIV
E,INTERNAL,FIXED ORL,OUTPUT

```

check for and EOF (End Of File). Line 490 INPUTs a record A\$, and displays it on the screen. Line 500 checks to see if it is part of the tab settings used by TIW/FW, and if so goes to 503, else line 510 PRINTs it to the output file. Line 520 numbers the records (N) and displays the number as indicated by N, this not necessary, but nice. Then it returns to 480 to check EOF and read another record. When EOF is reached, the read cycle is exited to line 530. The files are closed and the program returns to line 120. This is the END of the program.

```

AS=CHR$(Q)
1030 DISPLAY AT(ROW,1)SIZE(30):" You Selected ";AS
:: SUBEND
1040 SUB BEEP :: CALL SOUND(100,1250,0):: SUBEND
1050 SUB CLS(R1,R2):: FOR R=R1 TO R2 :: DISPLAY AT(R,1)SIZE(30):: NEXT R :: SUBEND
1060 SUB AGAIN :: DISPLAY AT(24,1):"Again? Press A, Else Any Key"
1070 CALL KEY(3,K,S):: IF S<1 THEN 1070 ELSE IF K<>ASC("A")THEN RUN "DSK1.DIR"
1080 SUBEND
1090 SUB GO(A$,K)
1100 CALL GKEY(Q,22):: K=POS(A$,CHR$(Q),1):: IF K=0 THEN 1100
1110 SUBEND
1120 SUB PAK
1130 DISPLAY AT(24,1)SIZE(30):" Press Any Key to Proceed"
1140 CALL KEY(0,K,S):: IF S<1 THEN 1130
1150 DISPLAY AT(24,1)SIZE(30):: SUBEND
1160 SUB READIT(Z)
1170 N=1 :: CALL CLEAR
1180 IF EOF(Z)THEN 1200 ELSE INPUT #2:AS :: DISPLAY AT(N,1):AS :: N=N+2 :: IF N>22 THEN CALL PAK :: GOTO 1170
1190 GOTO 1180
1200 DISPLAY AT(23,9):"End Of File " :: CALL PAK :: SUBEND

```

```

:: RETURN
780 OPEN #2:ODN$&OF$,RELATIVE,INTERNAL,FIXED ORL,OUTPUT
:: RETURN
790 OPEN #2:ODN$&OFS$,RELATIVE,DISPLAY ,FIXED ORL,OUTPUT
:: RETURN
800 OPEN #2:ODN$&DFS$,RELATIVE,DISPLAY ,FIXED ORL,OUTPUT
:: RETURN
810 ! **select input file**
820 ON IO+IT+IK+1 GOSUB 620,640,650,660,670,680,690,700
830 RETURN
840 ! **select output file**
850 ON OO+OT+OK+1 GOSUB 730,740,750,760,770,780,790,800
860 RETURN
870 DISPLAY AT(1,1)ERASE ALL:" FILE CONVERSION PROGRAM"
880 ! **select input file**
890 DISPLAY AT(3,1):"This program allows you to access any file, if you can give its characteristics."
900 DISPLAY AT(7,1):"You can read it or write thefile info to almost any other kind of file."
910 DISPLAY AT(11,1):"You may specify REL,SEQ,DIS,INT,UAR or FIX. Note that if you specify REL, you will"
920 DISPLAY AT(14,1):"default to FIX record length"
930 DISPLAY AT(16,1):"You must specify a record length. You also must use all title sense, don't write a DF 255 file to an IU20 for example."
940 DISPLAY AT(22,1):"You can't read PROGRAM files"
950 CALL PAK :: GOTO 120
960 DISPLAY AT(12,1)ERASE ALL:"What is the Record length 80" :: ACCEPT AT(12,26)SIZE(-3):RL :: RETURN
970 CALL GKEY(K,24):: IF K=ASC("R")THEN 870 ELSE 120
980 !
990 ! SUBPROGRAM AREA
1000 !
1010 SUB GKEY(Q,ROW)
1020 CALL KEY(3,K,S):: IF S<1 THEN 1020 ELSE Q=K :: IF Q=32 THEN AS="Space" ELSE

```

## TI WRITER TIP From Nutmeg News

Here is a tip for TI-Writer users. Some of you are probably familiar with the Replace String (RS) function, however, here are a few tricks to its use that can increase its effectiveness for you.

1. Before using RS, make sure to turn off the word wrap mode. Do this by pressing FCTN 0 until you get a hollow cursor. This will prevent TIW from reformatting the whole document. The only time you would want this type of reformatting is when you are replacing a short string with a much longer one, e.g. "FW" replaced by "Funnelweb Farm Utility Disk". If you must leave word wrap on because of this, remember that each paragraph must end in a carriage return, and any line that has special spacing or indentations should end in a carriage return as well.

Try this on something like a doc file from a fairware disk. Load in a large file, press FCTN 9, and then type RS, E(nter). Now type / a / q / and press E(nter). Yes, there are spaces before and after the letters "a" and "q", and both letters should be framed by slashes as shown. TIW will find the first occurrence of "a", and then will ask "Yes No All Stop". Typing "Y(es)" and E(nter) will

change just this "a" to a "q" if you are in fixed mode (hollow cursor), however if you are in word wrap mode TIW will replace the "a" and then reformat the rest of the paragraph. Take this one step further and type "A(11)", E(nter). This will change every occurrence of "a" to "q". Time this and then hit FCTN 0 and change "q" back to "a" by changing the RS string to / q / a /. You should find the fixed mode to be many times faster than the word wrap mode.

2. RS is sensitive to columns, meaning that you do not have to have every word in the document checked if you want to change "2]" to "2)". Use this string to replace a string occurring between columns 8 and 10:

8 10 /]//

Remember, always back up your files and work with the backup. Doing some of these exercises can be fun, but only if there is no risk involved. Until next time, keep well and enjoy.

[Ed note: Reprinted from PUG Peripheral, July 92. The above procedure will work equally well with FunnelWriter.]

### 24K OF DATA STORAGE

Author Unknown

If you need to work with quite a bit of data or would like to change programs but save the data after you press CALL QUIT, then you can set up the 24K of High-Memory in the PEB as a single data file called "EXPHEM2". You can open this file just as you would any file with one exception - you must precede the OPEN statement with a CALL LOAD to location -24574 as follows:

For INT/VAR files - 24  
For DIS/VAR files - 16  
For INT/FIX files - 8  
For DIS/FIX files - 0

Here is an example:

If you want to open the expansion memory for D/V 80 files, this is what you do.

```
100 CALL INIT
110 CALL LOAD(-24574,-16)
120 OPEN 81:"EXPHEM2", RELATI
VE,UPDATE,DISPLAY,VARIABLE 80
```

Then continue as you normally would.

If you wish to store both data and assembly language routines at the same time, do this:

```
100 CALL INIT
110 CALL LOAD(-24574,-16)
120 OPEN 81:"EXPHEM2"
130 CALL LOAD("DBK1.ASEM1")
140 CALL LOAD("DBK1.ASEM2")
150 CALL LINK("START")
```

### 160 REM CONTINUE REST OF PROGRAM

In the above example, the 24K of high memory was saved for use as a DATA file (D/V 80 format). Then the assembly routines were loaded. The computer will look for the best place to put the routines and will adjust the pointer accordingly. After the routines are loaded, a link statement starts the first routine and off we go.

If that's not enough for you, you can also use the mini-memory for 4K more of assembly routine storage! Now that's 18K of program space, 12K for assembly routines.

[Ed note: Reprinted from ROM newsletter, Feb 91 through PUG Peripheral newsletter, July 92.]

TI WORLD NEWS February 1992  
compiled by Jim Peterson

The Long Island IIG newsletter for Jan. 1992 reproduces an ad for the ImageWise Serial Video System, in kit form, consisting of the digitizer/transmitter and the receiver/display. It is stated to capture an image in 1/60 of a second from monochrome or color video cameras, camcorders, VCRs, etc. According to the ad, with additional software it will digitize images for display on IBM PC, Apple Macintosh, Commodore Amiga, Atari ST and other popular computer systems. A handwritten note indicates that it is compatible with the TI-99/4A with 80 column card, or the Geneve 9640. The address is Micromint, Inc., 4 Park St., Vernon CT 06066, phone (203) 875-2751. Price is not mentioned.

The same newsletter contains ads for TI-99/4A software to be used with the ImageWise system. These include a \$9.95 disk (plus \$1.50 S&H) containing the programs GRAB, SHOW and CONVERT (for use with the TI-99/4A; no mention that an 80-column card is required); ImageWise Portrait Print (\$4.95 + \$1.50 S&H) to print a 17"x22" poster from a digitized file; ImageWise Display Routine (\$4.95 + \$1.50 S&H) for a TI-99/4A equipped with the Yamaha 9938 VDP, or the Geneve 9640; ImageWise pictures in two volumes: all available from R.F.W. Enterprises, 111 Oakridge Street, Chicopee MA 01020.

And, available from Joseph M. Syzdek, 99 Highland Ave., West Springfield MA 01089-1017 for \$14.95 plus \$1.50 S&H, is IWD Plus for the Geneve 9640 and ImageWise Video Digitizer, to capture digitized video data over the RS232 port and display the image on a monitor in 256x212 or 512x212 resolution. It can be saved to disk in ImageWise or MyArt format and provides some editing capability.

Harrison Software is now offering a 45-minute stereo cassette of their MIDI-Master music, containing the 20 pieces from Magdalena's Notebook by J.S. Bach and Bach's Italian Concerto. Except for the last, all were produced on a CASIO CT-650 keyboard using a TI-99/4A with MIDI-Master 99. The price is \$10 and the address is 5705 40th Place, Hyattsville MD 20781.

Harrison Software has announced that they are removing their assembly music disks from their software catalog, and placed them in the public domain. User groups may distribute them as they wish. They will also be available from Tigercub Software, 156 Collingwood Ave., Columbus OH 43213 (\$1.50 per disk, plus \$1.50 shipping and handling if less than 8 disks are ordered).

I have a letter from Francisco T. Molina, a really isolated TI enthusiast in Argentina, who is trying to organize a local user group to be called T.I.G.R.E.S. de Argentina, which stands for Texas Instruments (99/4A) Grupo Recalcitrante y Espernido de Sobrevivientes, meaning "group of everlasting, recalcitrant survivors". He reports that his local mail service is quite unreliable but he has now found a friend in Virginia who apparently makes frequent trips to Argentina and can carry software to him.

For those who like "brain games", Asgard Software (P.O. Box 10306, Rockville MD 20849) has released TI PEI, a mahjonn game by William Reias, and Classic Checkers by Chris Bobbitt. They load from disk and require XBasic and 32k; a Mouse is optional. The price of each is \$14.95 plus \$3 per order for S&H (\$3.50 in Canada).

### 3 1/2 INCH DISK DRIVES ON THE TI99/4A

BY FRANK AYLSTOCK

The 5.25" (360k) drives are becoming another orphan, like our TI. The disk controllers do not know if you have 3.5" or 5.25" drives. The only thing they know is what your input is, and the only control you have is the number of tracks per sector, number of sides and density. The TI disk controller will handle double sided but only single density. The Corcomp controller will handle double sided and double density. The Myarc card with the QUAD CHIP installed will handle disk drives up to 720k. The 5.25" quad density drives are another orphan but you can use 3.5" disk drives. The 3.5" drives can be up to 1.44meg this means that you will have 2880 sectors or the equivalent of 8 5.25" floppy disks. The only drawback to the 3.5" drive is that all the programs you receive come on 5.25" floppy disks. However you can set up your system so that you have at least one 5.25" disk drive and the others 3.5" drives. The HFDC by Myarc will also accept up to quad density disks.

I would recommend that you switch over to the 3.5" drives as they are a superior form of storage for the following reasons.

1) The disks are enclosed in a shell/cover which hold them rigid and will not allow the disk to get bent. You can even write on the disk directly without harming the data.

2) They contain their own sliding reusable "write protect tab". By merely moving the tab up or down the disk can be protected.

3) The size is a large consideration as they require a lot less space to store or transport them.

4) They contain a sliding door which protects the storage medium at all times. This door opens and closes automatically when the disk is inserted or removed from the drive.

5) The size also helps to read and write data faster than the 5.25" floppy disk drives.

6) The disks are coated with superior oxide which is less vulnerable to data loss.

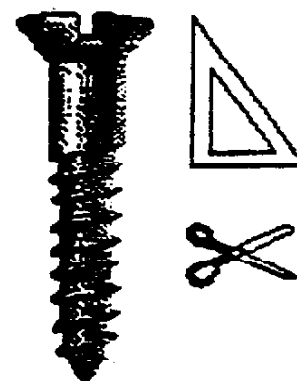
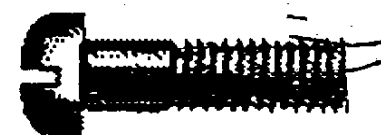
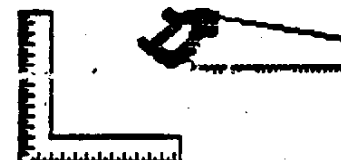
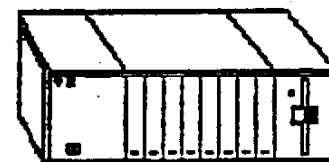
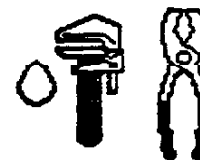
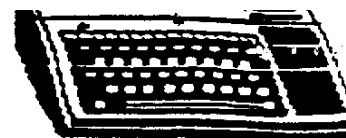
7) They are considered more reliable than 5.25" disks especially important when dealing with quad density disks.

8) The drives take less current during the reading and writing process. In fact some of the 3.5" drives use only the 5 volts.

Last but not least is the price. Around this area (Los Angeles) the drives can be purchased for as little as \$50.00 and there is no conversion or other hardware changes to be made and they will replace the existing drives with very little labor.

Look into these drives!

The above was reprinted from the BREA USERS GROUP





LA99 USER GROUP LIBRARY 12/1/92

SY SILVER (LIBRARIAN)  
4162 W.172ND ST.  
TORRANCE, CA 90504  
(310) 370-2145

"HOLIDAY SALE" for DECEMBER. "HAPPY HOLLIDAY". All disks with programs are reduced to \$1.00. This includes the disk. A few samples disks in some of the categories will be displayed below. Remember, there are many and possibly more useful programs for you in the library. Check the library catalogue for the many other excellent programs at this discount price.

UTILITY DISK SAMPLE

2612 MASS TRANSFER Fairware by Stuart Olsen. An assembly language Terminal Eliminator program for modem use, menu drive X.Y. Modem capable of multiple transfer all at once. Auto dial, upload, down load multiple RAM disks. Many features. (DSSD)(407) 2612A(146) and 2612B(257) for SSSD drives.

GAME SAMPLES (Why pay \$35 or more  
for game programs in other systems)

9062 TI-CRAPS Play against the house just as one would in a casino and test your system before going to a legal casino. I found that laying a bet on the pass line, then taking the odds on the pass number, the six and the eight gives very close to even odds. With that method a player can come out ahead with reasonable luck. The computer is the banker and keeps track of your money account. SSSD (302)

9053 GAMES 53 BACKGAMMON. It plays well, but you can win if you play well and if the dice random numbers come up in your favor. Also VIDEO POKER. "Why go to Las Vegas, when you can play at home and never lose capital." Also on the disk, RUBICSCUBE "FUN"

MUSIC SAMPLE (HOLIDAY MUSIC)

7035 MUSIC #35 Holiday music 20 programs, Xmas, Valentine's Day, 4th of July, Halloween, Birthday, Party, ETC, Basic and E/X, E/A, Printer sssd(280)

Other categories to check are: Disk Management 26--; Utility 28--; Business 30--; Misc. 40--; Education 50--; Home 60--; Music 70--; Drawing 80-- etc.

# LA 99er's OFFICERS

PRESIDENT.....	Edgar May.....	310 <del>326 6660</del> <sup>644-6241</sup>
V.PRESIDENT.....	Earl Raguse .....	714 847 5875
Treasurer.....	Al Whiteman .....	310 538 1953
Secretary.....	Bob Wheeler.....	310 378 5092
Editor.....	Joe Fierstein .....	310 377 9834
Membership .....	Hal Jeffery .....	310 372 7795
Equipment .....	Gail Fair .....	310 326 6660
Program .....	Steve Mehr .....	805 379 2937
Market Place .....	Fred Moore .....	310 670 4293
User Group .....	Fred Moore .....	310 670 4293
Library .....	Sy Silver .....	310 370 2145

LA 99ers  
 A TI User Grp  
 7738 Emerson Ave  
 Los Angeles, Co. 90045

29 USA



Expiring 4 Sep 92  
 DALLAS TI COMPUTER GROUP  
 P. O. Box 29863  
 Dallas TX 75209